

**WHAT IS CLAIMED IS:**

1. A system comprising:
  - a persistence layer to store a plurality of properties in a computer-readable storage medium;
  - a property manager layer to retrieve from the persistence layer values of those of the properties used by a first software application and to provide the values to the first software application at run time;
  - a first property listener to modify values of program variables in the first software application that are dependent on the properties without having to restart the first software application;
  - a user interface layer to accept user input to update at least one of the properties, wherein the persistence layer is responsive to the user interface layer to update the at least one of the properties in the computer-readable storage medium based on the user input, and wherein the user interface layer is to notify the property manager layer that the at least one of the properties has been updated;
  - wherein the property manager layer is to determine if any of the at least one of the properties that has been updated is used by the first software application, and to notify the first property listener of at least one updated property value used by the first software application based thereon; and
  - wherein the first property listener is to modify at least one program variable in the first software application that is dependent on the at least one updated property value without restarting the first software application.
2. The system of claim 1 further comprising:
  - a second property listener to modify values of program variables in the second software application that are dependent on the properties without having to restart the second software application;
  - wherein the property manager layer is to determine if any of the at least one of the properties that has been updated is used by the second software

application, and to notify the second property listener of at least one updated property value used by the second software application based thereon; and

wherein the second property listener is to modify at least one program variable in the second software application that is dependent on the at least one updated property value without restarting the second software application.

3. The system of claim 2 wherein each of the properties is identified by an associated property key, and wherein the property manager layer comprises a lookup table to record, for each of the property keys, a list of property listeners that have registered for the property key.

4. The system of claim 2 wherein the first software application comprises a first application class of a middleware application, and wherein the second software application comprises a second application class of the middleware application.

5. The system of claim 4 wherein the middleware application provides flow through from a plurality of ordering applications to at least one back-end application.

6. The system of claim 4 wherein the property manager layer is contained within a common project within the middleware application.

7. The system of claim 4 wherein the property manager layer is contained in a client JAVA Archive (JAR) file.

8. The system of claim 4 wherein the property manager layer is contained in a Dynamic Link Library (DLL) file.

9. The system of claim 1 wherein the persistence layer is to store information associated with the properties.

10. The system of claim 9 wherein, for each of at least one of the properties, the information comprises a version number, a time stamp, an author name, and a description associated therewith.

11. The system of claim 9 wherein the user interface layer is to display a name, a value, and the information associated with at least one of the properties.

12. The system of claim 11 wherein the user interface layer is to accept user input to update the value associated with a property.

13. The system of claim 11 wherein the user interface layer is to accept user input to update the information associated with a property.

14. The system of claim 11 wherein the user interface layer further is to display a time stamp associated with the at least one of the properties.

15. The system of claim 1 wherein the user interface layer accepts user input to add a property, to modify a property, and to delete a property.

16. The system of claim 1 wherein the user interface layer is to organize and display the properties in a plurality of categories.

17. The system of claim 1 wherein the user interface layer comprises a Web interface layer.

18. The system of claim 1 wherein the user interface layer is to display the properties in a searchable format.

19. A method comprising:  
storing a plurality of properties in a computer-readable storage medium;  
retrieving, from the computer-readable storage medium, values of those of the  
properties used by a first software application and to provide the values to  
the first software application at run time;  
accepting user input to update at least one of the properties;  
updating the at least one of the properties in the computer-readable storage  
medium based on the user input;  
based on the user input, determining if any of the at least one of the properties that  
has been updated is used by the first software application;  
notifying a first property listener of at least one updated property value used by  
the first software application; and  
modifying, by the first property listener, at least one program variable in the first  
software application that is dependent on the at least one updated property  
value without restarting the first software application.

20. A computer-readable medium having computer-readable program code to cause a computer system to perform acts of:

- storing a plurality of properties in a computer-readable storage medium;
- retrieving, from the computer-readable storage medium, values of those of the properties used by a first software application and to provide the values to the first software application at run time;
- accepting user input to update at least one of the properties;
- updating the at least one of the properties in the computer-readable storage medium based on the user input;
- based on the user input, determining if any of the at least one of the properties that has been updated is used by the first software application;
- notifying a first property listener of at least one updated property value used by the first software application; and
- modifying, by the first property listener, at least one program variable in the first software application that is dependent on the at least one updated property value without restarting the first software application.